CENTRE DE RECHERCHE
SUR LES TRANSPORTS
(C.R.T.)
CENTRE FOR RESEARCH
ON TRANSPORTATION

UNIVERSITÉ DE MONTRÉAL
ÉCOLE DES HAUTES ÉTUDES COMMERCIALES
ÉCOLE POLYTECHNIQUE

CENTRE DE RECHERCHE SUR LES TRANSPORTS
UNIVERSITÉ DE MONTRÉAL
C.P. 6128, SUCCURSALE A
MONTRÉAL
CANADA H3C 3J7

TÉLÉPHONE : (514) 343-7575
EMAIL : crt@crt.umontreal.ca
TÉLÉCOPIEUR (FAX) : (514) 343-7121

93-01753/

CRT-684

Vehicle Routing with Split Deliveries*

by
Moshe Dror[1,2]
Gilbert Laporte[2]
Pierre Trudeau[2,3]

[1] Decision Sciences, College of Business, University of Arizona, Tucson, AZ 85721, U.S.A.
[2] Centre de recherche sur les transports, Université de Montréal, C.P. 6128, succursale A, Montréal, Québec, Canada H3C 3J7.
[3] Ad Opt, 4475 boulevard Saint-Laurent, Montréal, Québec, Canada H2W 1Z8.

* The authors are grateful to the Canadian Natural Sciences and Engineering Research Council for its financial support (grants OGP 0036429 and OGP 0039682). Thanks are also due to Martin Desrochers and to the referees for their helpful suggestions.

Centre de recherche sur les transports — Publication #684

February 1990 – Revised April 1992

Dépôt légal – 2e trimestre 1992

ABSTRACT

This paper considers a relaxation of the classical vehicle routing problem (VRP), in which split deliveries are allowed. As the classical VRP, this problem is NP-hard, but nonetheless it seems more difficult to solve exactly. It is first formulated as an integer linear program. Several new classes of valid constraints are derived, and a hierarchy between these is established. A constraint relaxation branch and bound for the problem is then described. Computational results indicate that by using an appropriate combination of constraints, the gap between the lower and upper bounds at the root of the search tree can be reduced considerably. These results also confirm the quality of a previously published heuristic for this problem.

Key words: split delivery vehicle routing problem, subtour elimination constraints, connectivity constraints, k-split cycles, fractional cycle elimination constraints.

RÉSUMÉ

On considère dans cet article une relaxation du problème classique de tournées de véhicules dans lequel on permet des demandes fractionnables. Comme le problème de tournées classique, ce problème est NP-difficile, mais il semble néanmoins plus difficile à résoudre de façon exacte. On le formule d'abord comme un programme linéaire en nombres entiers. On dérive ensuite plusieurs nouvelles classes de contraintes valides pour ce problème et l'on établit une hiérarchie entre elles. Vient ensuite la description d'un algorithme pour ce problème, basé sur la relaxation de contraintes et sur une technique de branch and bound. Les résultats numériques indiquent qu'en utilisant une combinaison judicieuse des contraintes, on parvient à réduire de façon considérable l'écart entre la borne inférieure et la borne supérieure à la racine de l'arbre de branch and bound. Ces résultats confirment également la qualité d'une méthode heuristique antérieure développée pour ce problème.

Mots-clés: problèmes de tournées avec demandes fractionnables, contraintes d'élimination de sous-tours, contraintes de connectivité, "k-split-cycles", contraintes d'élimination de cycles fractionnaires.

1. Introduction

The classical *Vehicle Routing Problem* (VRP) can be defined as follows. Let $G = (N, A)$ be a graph where $N = \{0, ..., n\}$ is a set of vertices corresponding to cities, and $A = \{(i, j) : i, j \in N, i \neq j\}$ is the arc set. Vertex 0 represents a *depot* at which a fleet of m vehicles is based; the remaining vertices correspond to *customers*. In general, m belongs to some interval $[\underline{m}, \overline{m}]$,, where $1 \leq \underline{m} \leq \overline{m} \leq n$. Vehicles may have equal or different capacities. In general, let vehicle v have a capacity equal to Q_v. Every vertex i of $N \setminus \{0\}$ has a non-negative demand $q_i \leq \max_v \{Q_v\}$ and every arc (i, j) has an associated non-negative distance or travel cost c_{ij}. The VRP consists of determining a set of minimum cost vehicle routes:

(i) starting and ending at the depot;

(ii) such that every customer is visited once by one vehicle and

(iii) such that the total demand of any route does not exceed the capacity of the vehicle assigned to that route.

It is well known that the VRP is NP-hard since when $m = 1$ and $Q_1 \geq \sum_{i=1}^{n} q_i$, it then reduces to the *Travelling Salesman Problem* (TSP). There exists an abundant literature on the VRP and related problems. For recent surveys on algorithms, see Christofides [4], Laporte and Nobert [15], and Laporte [13]. For results on worst case behaviour of some heuristic algorithms, see Altinkemer and Gavish [1,2]. Recently, Dror and Trudeau [7,8] have investigated a relaxation of the VRP in which condition (ii) is removed, i.e., customer demand can be split between several vehicles. In this

context, it is no longer necessary to assume that $q_i \leq \max_v \{Q_v\}$. This variant of the VRP is called the *Split Delivery Vehicle Routing Problem* (SDVRP). Dror and Trudeau [7,8] have proposed a heuristic algorithm for the SDVRP and have shown that allowing split deliveries can yield substantial savings, both in the total distance travelled and in the number of vehicles used in the optimal solution. Unfortunately, the SDVRP is still NP-hard (Dror and Trudeau [8]).

The object of this paper is to provide an integer linear programming (ILP) formulation including new families of valid inequalities, as well as an exact constraint relaxation algorithm for the SDVRP. The paper is structured as follows. The ILP formulation is presented in Section 2. In Section 3, we provide a detailed discussion of subtour elimination and connectivity constraints in the context of the SDVRP. New classes of valid constraints are introduced in Section 4. The algorithm description is contained in Section 5, and computational results are reported in Section 6. The conclusion follows in Section 7.

2. Formulation

Let x_{ijv} be a binary variable defined for $i \neq j$ and equal to 1 if an only if in the optimal solution, vehicle v travels directly from i to j. Let y_{iv} be the proportion of the i^{th} customer demand delivered by vehicle v. The problem is then:

$$\text{minimize} \quad \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{v=1}^{\overline{m}} c_{ij}\, x_{ijv} \tag{1}$$

subject to:

$$\sum_{i=0}^{n} x_{ikv} - \sum_{j=0}^{n} x_{kjv} = 0 \qquad (k = 0, ..., n; v = 1, ..., \overline{m}) \quad (2)$$

$$\sum_{v=1}^{\overline{m}} y_{iv} = 1 \qquad (i = 1, ..., n) \quad (3)$$

$$\sum_{i=1}^{n} q_i y_{iv} \leq Q_v \qquad (v = 1, ..., \overline{m}) \quad (4)$$

$$\sum_{j=0}^{n} x_{ijv} \geq y_{iv} \qquad (i = 1, ..., n; v = 1, ..., \overline{m}) \quad (5)$$

Subtour elimination and connectivity constraints (6)

$$x_{ijv} \in \{0, 1\} \qquad (i, j = 0, ..., n; v = 1, ..., \overline{m}) \quad (7)$$

$$0 \leq y_{iv} \leq 1 \qquad (i = 1, ..., n; v = 1, ..., \overline{m}). \quad (8)$$

In this formulation, constraints (2) are flow conservation conditions. Constraints (3) specify that the demand of any customers is entirely satisfied. Constraints (4) ensure that vehicle capacities are never exceeded, while constraints (5) guarantee that if customer i is visited by vehicle v, then the same vehicle leaves that customer. *Subtour elimination constraints* (6) require a more elaborate discussion and will be described in Section 3. Note that summing up constraints (5) over all vehicles yields the following *connectivity constraints*:

$$\sum_{v=1}^{\overline{m}} \sum_{j=0}^{n} x_{ijv} \geq \sum_{v=1}^{\overline{m}} y_{iv} = 1 \qquad (i = 1, ..., n), \quad (9)$$

implying that any customer i will receive at least one visit.

3. Subtour Elimination and Connectivity Constraints

In this section, we provide valid subtour elimination and connectivity constraints for the SDVRP, and compare them with similar constraints developed for the classical VRP.

Subtour elimination constraints for the SDVRP are derived from the corresponding constraints for the TSP and for the VRP. First consider the standard subtour elimination constraints introducted by Dantzig *et al.* [5] for the TSP:

$$\sum_{i,j \in S} x_{ij} \leq |S| - 1 \qquad (S \subset N \setminus \{0\}; 2 \leq |S| \leq n-1). \quad (10)$$

These constraints eliminate all subtours defined over subsets of $N \setminus \{0\}$ containing between 2 and $n-1$ vertices. Since in the TSP there is only one vehicle, x_{ij} must be interpreted as x_{ij1} in the SDVRP formulation. It is straightforward to show that subtour elimination constraints (10) are equivalent to the following connectivity constraints:

$$\sum_{i \in S, j \in \overline{S}} x_{ij} \geq 1 \qquad (S \subset N \setminus \{0\}; 2 \leq |S| \leq n-1), \quad (11)$$

where $\overline{S} = N \setminus S$.

In the case of the classical VRP, constraints (10) can be strenghtened to:

$$\sum_{v=1}^{\overline{m}} \sum_{i,j \in S} x_{ijv} \leq |S| - V(S) \qquad (S \subseteq N \setminus \{0\}; |S| \geq 2), \qquad (12)$$

where $V(S)$ is the number of vehicles required to serve all nodes of S in any feasible VRP solution (see, e.g., Laporte and Nobert [14], Gavish [10,11] and Laporte *et al.*

[16]). The value of $V(S)$ can be determined by solving a bin packing problem [10], but a lower bound is often used. In this paper constraints (12) are initially relaxed and successively introduced. At a given solution, $V(S)$ is obtained by first determining $W(S) = \{v : x_{ijv} > 0, i \in S, j \notin S\}$, and $V(S)$ is the smallest number of vehicles of $W(S)$ necessary to cover the total demand of S. Again, the equivalence of (12) with the following connectivity constraints is immediate:

$$\sum_{v=1}^{\overline{m}} \sum_{i \in S, j \in \overline{S}} x_{ijv} \geq V(S) \qquad (S \subseteq N \setminus \{0\}; \mid S \mid \geq 2). \qquad (13)$$

Constraints (12) or (13) eliminate two types of infeasibilities: (i) subtours disconnected from the depot and (ii) vehicle routes connected to the depot, but whose total demand exceeds the vehicle capacity.

Observe that constraints (12) are invalid for the SDVRP. Indeed, consider the example in Figure 1, with $S = \{1, 2, 3, 4, 5\}, Q_v = 3$ for all $v, q_1 = q_2 = q_3 = q_4 = 1, q_5 = 2$. Here, the minimum number of vehicles required to satisfy the demand of S is $V(S) = 2$. Sharing the demand of customer 5 equally between the two vehicles yields the feasible solution shown in Figure 1. However, constraint (12) applied to S is not satisfied since its left-hand side is equal to 4, while its right-hand side is equal to 3.

It is straightforward, however, to prove that constraints (13) are still valid for the SDVRP. This apparent contradiction is explained by the fact that the equivalence between (12) and (13) holds as long as the incoming (or outgoing) degree of vertex of S is equal to 1 in the optimal solution (see Laporte [12] for a discussion of this

property in the case of undirected graphs). This clearly is not the case in the SDVRP. However, a valid equivalence can still be derived. Let d_i denote the outgoing degree of vertex i:

$$d_i = \sum_{v=1}^{\overline{m}} \sum_{j=0}^{n} x_{ijv} \qquad (i \in N). \quad (14)$$

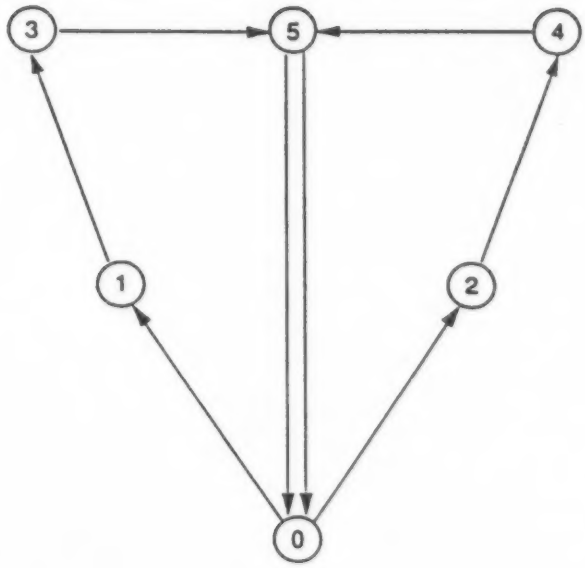

Figure 1 : Counter-example showing that the classical VRP subtour elimination constraints (12) are invalid for the SDVRP.

We then prove the following equivalence.

Proposition 1

The constraints

$$\sum_{v=1}^{\overline{m}} \sum_{i,j \in S} x_{ijv} \leq \sum_{i \in S} d_i - V(S) \qquad (S \subseteq N \setminus \{0\}; |S| \geq 2) \quad (15)$$

are equivalent to constraints (13) and are therefore valid inequalities for the SDVRP.

Proof

The result follows immediately from the fact that for any non-empty subset S of $N \setminus \{0\}$, the following relationship is true by definition of d_i:

$$\sum_{i \in S} d_i = \left(\sum_{v=1}^{\overline{m}} \sum_{i,j \in S} x_{ijv} + \sum_{i \in S, j \in \overline{S}} x_{ijv} \right). \qquad \blacksquare \quad (16)$$

In the SDVRP, constraints (15) can be imposed to eliminate subtours disconnected from the depot. Vehicle routes connected to the depot but having a total demand exceeding the vehicle capacity are prevented by (4).

Another concept closely related to subtours is that of *k−split cycles*.

Definition 1

Let $S = \{i_1, ..., i_k\} \subseteq N \setminus \{0\}$ and $k > 1$. If there exist h vehicle routes such that i_1 and i_k are on the same route, and that for every $t = 1, ..., k-1, i_t$ and i_{t+1} are on the same route, then S is a k–split cycle. $\blacksquare$

The original Dror and Trudeau [8] definition of a $k-$split cycle was restricted to the case where $h = k$. This is, however, not necessary for our purpose and the

required properties hold if the same route includes, for example, several pairs of vertices of S. If $h = 1$, the $k-$split cycle corresponds to a standard one-vehicle subtour. If $h = k$, then every customer of S (the $k-$split cycle) is split between exactly two vehicles. In what follows, it is implicitly assumed that $1 \leq h \leq k$. Dror and Trudeau [8] have proved the following result:

Proposition 2

If (c_{ij}) satisfies the triangle inequality, there always exists an optimal SDVRP solution not containing $k-$split cycles. ■

Note that if the triangle inequality is always strictly satisfied (i.e., $c_{ik} < c_{ij} + c_{jk}$ for all i, j, k), then no optimal SDVRP solution contains a $k-$split cycle. It is obvious from Definition 1 that if $S = \{i_1, ..., i_k\}$ defines a $k-$split cycle, then there exists a set S' satisfying $S \subseteq S' \subseteq N \setminus \{0\}$ and such that once arc directions are removed, the elements of S' form a cycle. This is illustrated by Figure 2 (here $S' = \{1, 2, 3, 4, 5, 6\}$). Similarly any (undirected) cycle defined on a subset S of $N \setminus \{0\}$ trivially defines an $|S|-$split cycle. It is therefore valid to eliminate all cycles on $N \setminus \{0\}$. We have thus proved

Proposition 3

The constraints

$$\sum_{v=1}^{\overline{m}} \sum_{i,j \in S} x_{ijv} \leq |S| - 1 \qquad (S \subseteq N \setminus \{0\}; |S| \geq 2) \quad (17)$$

are valid inequalities for the SDVRP. ■

Note that (17), as well as all inequalities developed in this paper, hold even if $C = (c_{ij})$ does not satisfy the triangle inequality.

In any VRP solution, the number of positive x_{ijv} variables is equal to $n+m$. The following proposition shows that this number can be larger in the case of SDVRPs.

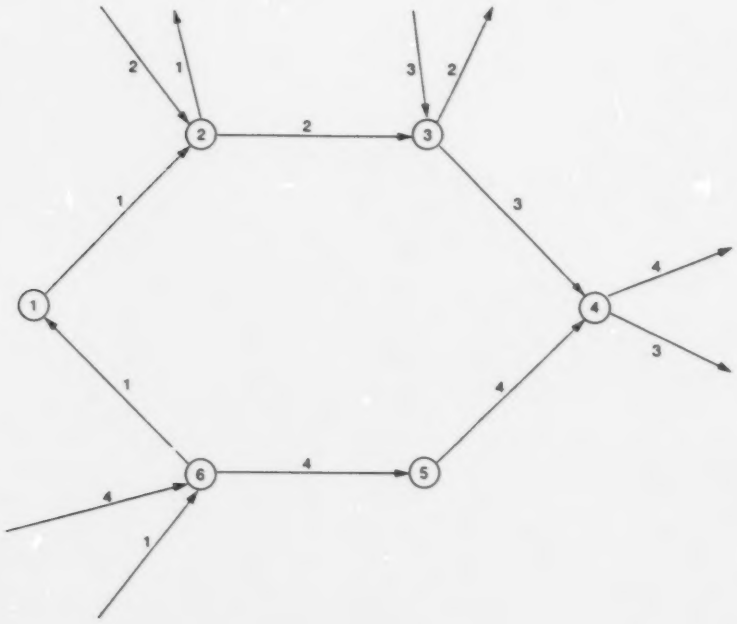

Figure 2: $S = \{2,3,4,6\}$ is a 4–split cycle. The values shown on the arcs are vehicle indices. When arc directions are removed, the element of S belong to the cycle formed by vertices 1,2,3,4,5 and 6.

Proposition 4

There always exists an optimal SDVRP solution in which the number of positive x_{ijv} variables is at most equal to $n+2m-1$. (In the case of strict triangle inequality, the number of positive variables is at most $n+2m-1$ in any optimal solution).

Proof

Every variable x_{ijv} with value 1 in the SDVRP solution corresponds to an arc in the solution graph. The maximal number of such arcs incident to the depot is $2m$. The number of arcs not incident to the depot cannot exceed $n-1$ (the number of arcs in a spanning tree over $N\backslash\{0\}$) since otherwise there would exist a $k-$split cycle on $N\backslash\{0\}$. ■

It is interesting to illustrate by means of a diagram the hierarchy between the various subtour elimination constraints developed for the TSP, the VRP and the SDVRP. The expression shown in Figure 3 are four possible right-hand sides for a subtour elimination constraint whose left-hand side is $\sum_{v=1}^{\overline{m}} \sum_{i,j \in S} x_{ijv}$. An arrow pointing from one constraint to another indicates that the former is stronger that the latter. As shown above, constraints (12) are valid for the VRP, but too strong for the SDVRP. Constraints (17') are valid for the SDVRP, but dominated by (15) and (17). In practice we use constraints (13) (equivalent to (15)) to eliminate subtours disconnected from the depot and situations in which the total demand of a set S of customers exceeds the total capacity of all vehicles assigned to S. In problems where (c_{ij}) satisfies the triangle inequality, we use (17) to eliminate $k-$split cycles. When $\overline{m} = 1$, the constraints reduce to the original TSP subtour elimination constraints.

4. Additional Classes of Valid Constraints

In addition to subtour elimination constraints and $k-$split cycle constraints, the following classes of constraints are also valid.

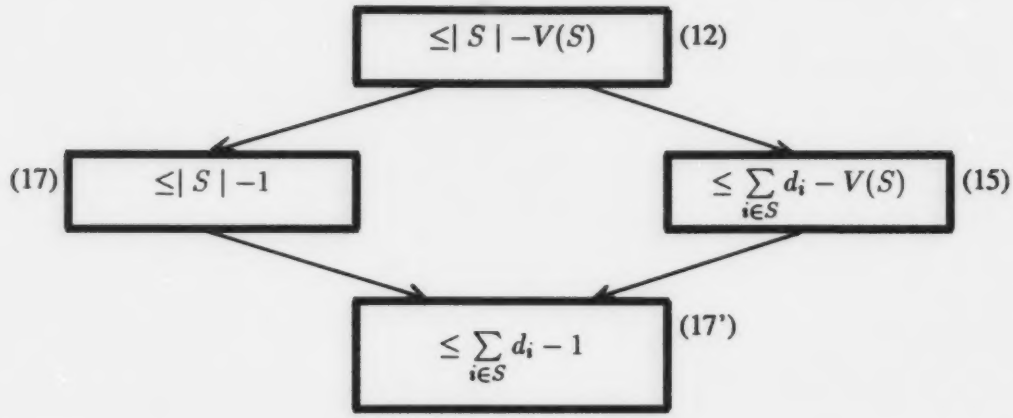

Figure 3 : Hierarchy between four types of subtour elimination constraints.

4.1 Outgoing degree of the depot

Since at least $\underline{m}$ vehicles are used in the solution, it is valid to impose

$$\sum_{j=1}^{n} x_{0jv} = 1 \qquad (v = 1, ..., \underline{m}) \qquad (18)$$

(assuming $Q_1 \geq Q_2 \geq ... \geq Q_{\overline{m}}$).

Similarly, x_{0jv} will be equal to 1 for at least one index j if vehicle v is used to visit a customer, therefore:

$$\sum_{j=1}^{n} x_{0jv} \geq \left(\sum_{i=1}^{n} q_i y_{iv} \right) / Q_{\underline{m}+1} \qquad (v = \underline{m} + 1, ..., \overline{m}). \qquad (19)$$

4.2 Variable fixing

When all vehicle have the same capacity, it is valid to assign one vehicle to one particular vertex i^*. We chose to assign vehicle 1 to the vertex i^* located the furthest away from the depot since this assignment has the most impact on the lower bound:

$$\sum_{j=0}^{n} x_{\bar{i}\bar{j}1} = 1. \tag{20}$$

In problems for which (c_{ij}) is symmetric, to avoid solutions which are merely symmetries of one another, it is valid to impose for one arbitrary pair of cities $(\bar{i}, \bar{j})$, the constraint

$$\sum_{v=1}^{\overline{m}} x_{\bar{i}\bar{j}v} = 0. \tag{21}$$

4.3 Fractional cycle elimination constraints I

Consider any non-empty subset S of $N \setminus \{0\}$ and vehicle v. If v does not visit any vertex of S, then $\sum_{i,j \in S} x_{ijv} = \sum_{i \in S, j \in \overline{S}} x_{ijv} = 0$. If v visits at least one vertex of S, then $\sum_{i,j \in S} x_{ijv} \leq |S| - 1$ and $\sum_{i \in S, j \in \overline{S}} x_{ijv} \geq 1$.
Thus, we have proved

Proposition 5

The constraints

$$\sum_{i \in S, j \in \overline{S}} x_{ijv} \geq \left(\sum_{i,j \in S} x_{ijv} \right) / (|S| - 1) \quad (S \subseteq N \setminus \{0\}; |S| \geq 2; v = 1, ..., \overline{m}) \tag{22}$$

are valid inequalities for the SDVRP. ■

A graphical interpretation of constraints (22) is provided by Figure 4. These constraints may sometimes be used to eliminate solutions containing fractional cycles. To illustrate, consider the example shown on Figure 5, with five nodes linked by four arcs traversed by the same vehicle. Suppose the values of the x_{ijv} variables are those

shown on the arcs. Then, constraints (22) are effective if S is defined as $\{1,2\}$ or as $\{3,4\}$.

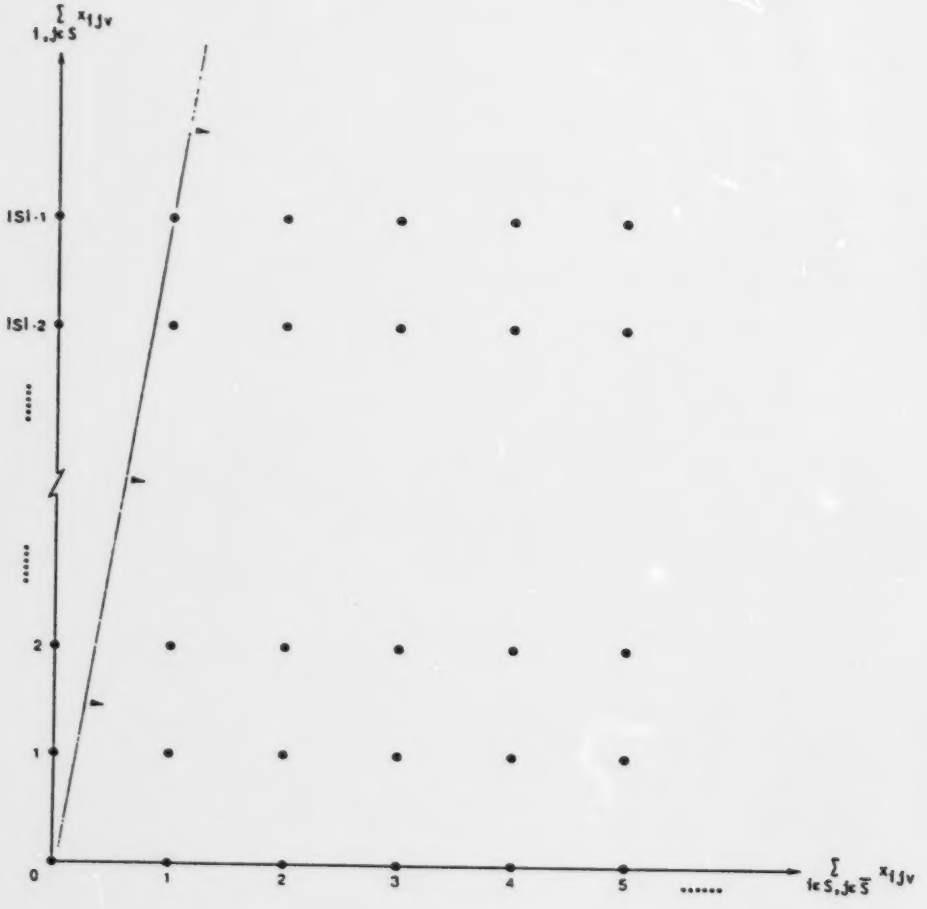

Figure 4 : Geometrical interpretation of constraints (22). Any feasible solution must lie below the line passing through (0,0) and $(1, |S|-1)$.

Figure 5: Fractional cycles eliminated by constraints (22)

Figures 6 and 7 illustrate what impact the introduction of a single constraint of type (22) can have on a solution. In Figure 6, we have represented a fractional solution to a 20 vertex problem, satisfying constraints (2)-(5), (8), and $0 \leq x_{ijv} \leq 1$ for all i, j, v (in this solution, all x_{ijv} variables take the value 1/2 unless otherwise indicated on the arcs). Figure 7 is obtained by applying the same constraints, as well as constraint (22) for the set $S = \{14, 19\}$. This results in a much simplified solution network, in which 2 vehicles are used instead of 5, and much closer to a feasible solution.

4.4 Fractional cycle elimination constraints II

The following proposition was suggested by Desrochers [6].

Proposition 6

The constraints

$$x_{ijv} \leq \sum_{k \neq i} x_{jkv} \qquad (i, j \in N \setminus \{0\}; v = 1, ..., \overline{m}) \quad (23)$$

are valid inequalities for the SDVRP.

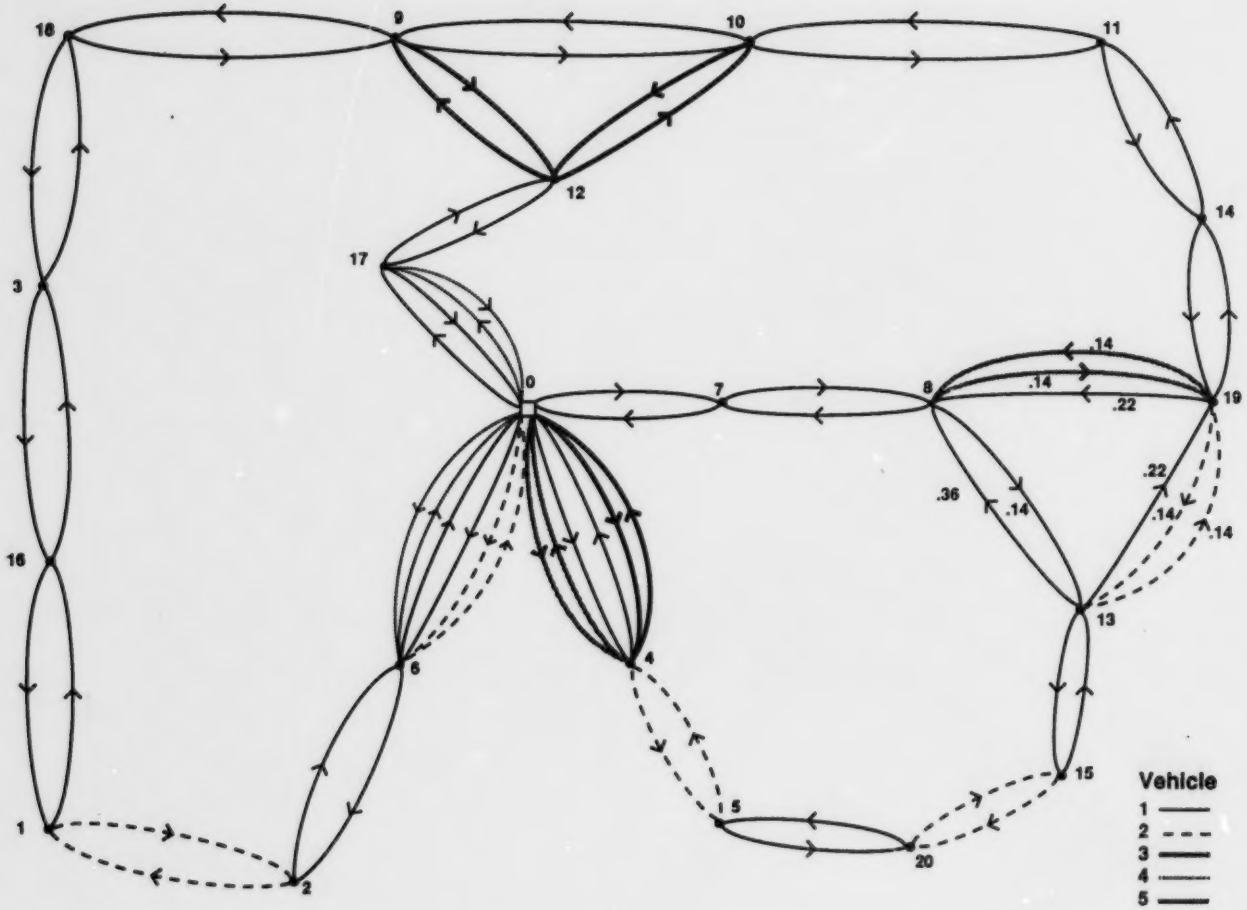

Figure 6: Solution corresponding to the initial linear relaxation.

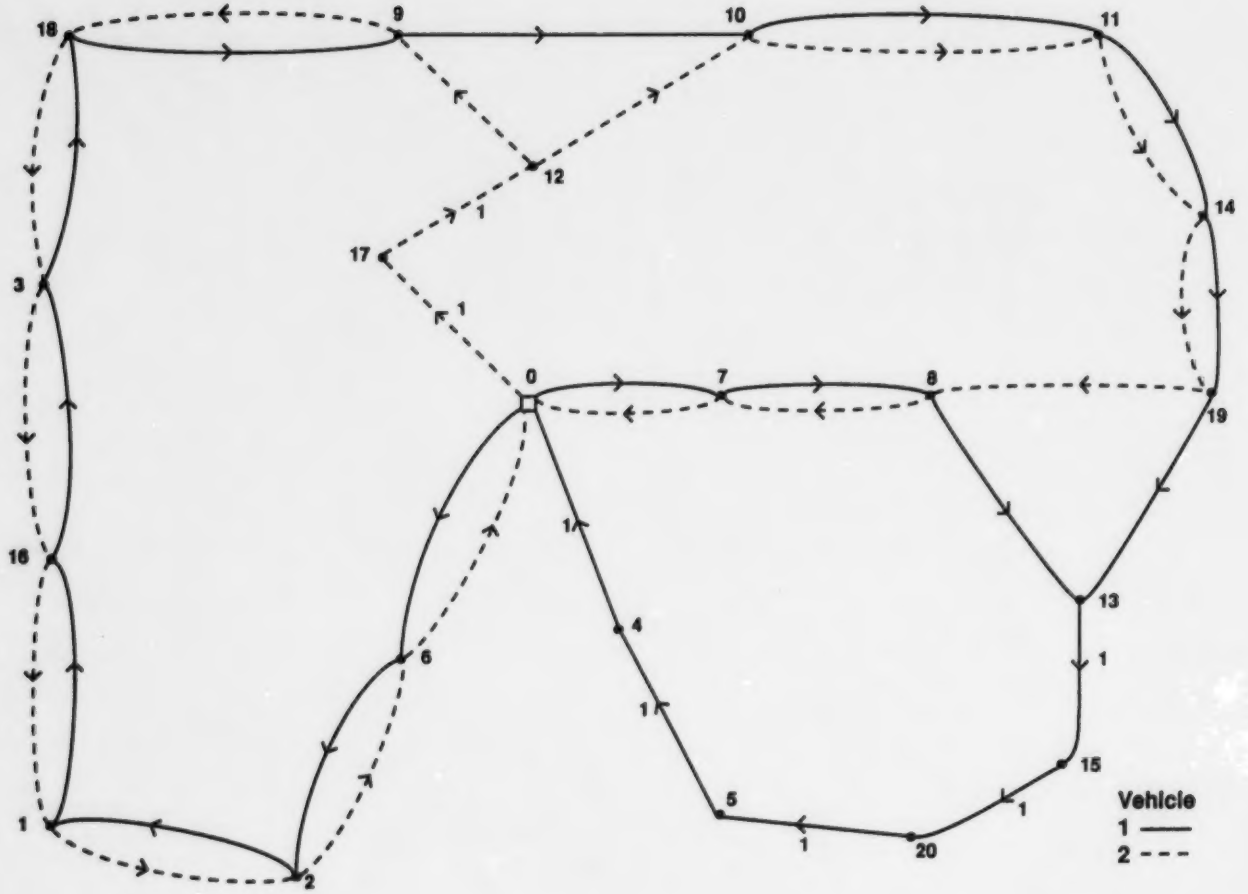

Figure 7: Solution obtained imposing constraint (22) for S = $\{14,19\}$.

Proof

For any $i, j \in N \setminus \{0\}$ and for any v, constraints (2) imply:

$$x_{ijv} \leq \sum_{k=0}^{n} x_{kjv} = \sum_{k=0}^{n} x_{jkv} = x_{jiv} + \sum_{k \neq i} x_{jkv}.$$

The conclusion follows immediately from the fact that $x_{ijv} = 0$ whenever $x_{ijv} = 1$. ■

There exist infeasible situations for which constraints (23) are violated while constraints (22) are satisfied. Consider, for example, the situation shown in Figure 8. Constraints (22) are satisfied for all pairs $(i, i+1)(i = 1, 2, 3)$ provided $\epsilon > 1/4$. However, constraint (23) is violated for $i = 3, j = 4$ since $x_{3,4,v} = 1 - 3\epsilon$ and $\sum_{k \neq 3} x_{4,k,v} = 0 < 1 - 3\epsilon$ when $\epsilon < 1/3$.

Another interesting relationship can be derived between constraints (22) and (23).

Proposition 7

For $S = \{\bar{i}, \bar{j}\} \subseteq N \setminus \{0\}$, constraints (22) are implied by constraints (23).

Proof

Summing up constraints (23) first defined for $i = \bar{i}$ and $j = \bar{j}$, and then for $i = \bar{j}$ and $j = \bar{i}$ yields $\sum_{k \neq \bar{i}} x_{\bar{j}kv} + \sum_{k \neq \bar{j}} x_{\bar{i}kv} \geq x_{\bar{i}\bar{j}v} + x_{\bar{j}\bar{i}v}$. This is precisely the expression of contraints (22) for $S = \{\bar{i}, \bar{j}\}$. ■

Using this result, a hierarchy can now be established between constraints (22)-(23), and those of Figure 3. These relationships are valid for any subset S of $N \setminus \{0\}$, except for the comparison of (22) and (23) which has been derived only for the case $|S| = 2$. The hierearchy is illustrated in Figure 9.

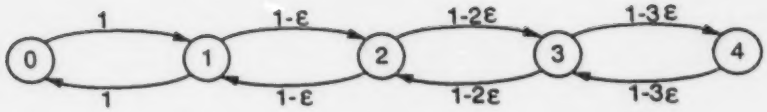

Figure 8 : Example for which constraints (23) are stronger than constraints (22).

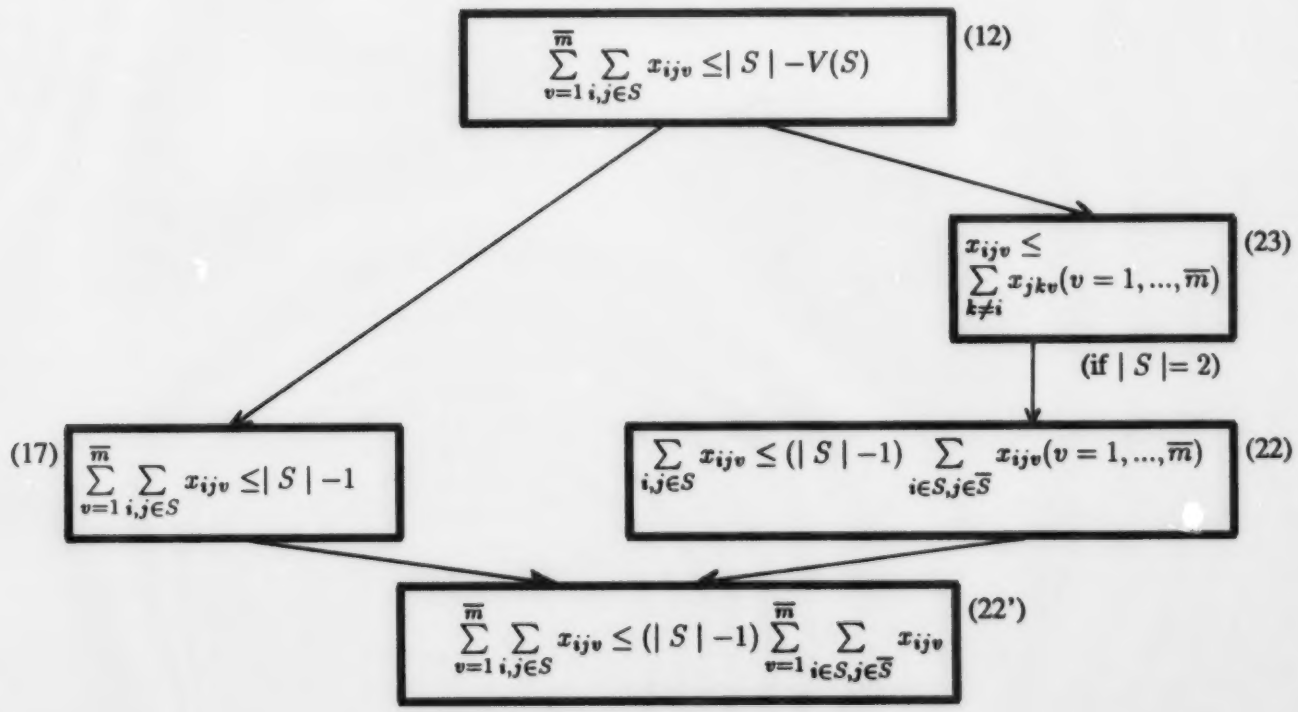
$\sum_{v=1}^{\overline{m}} \sum_{i,j \in S} x_{ijv} \leq |S| - V(S)$
(12)
$x_{ijv} \leq \sum_{k \neq i} x_{jkv} (v = 1, ..., \overline{m})$
(23)
(if $|S| = 2$)
(17)
$\sum_{v=1}^{\overline{m}} \sum_{i,j \in S} x_{ijv} \leq |S| - 1$
$\sum_{i,j \in S} x_{ijv} \leq (|S| - 1) \sum_{i \in S, j \in \overline{S}} x_{ijv} (v = 1, ..., \overline{m})$
(22)
$\sum_{v=1}^{\overline{m}} \sum_{i,j \in S} x_{ijv} \leq (|S| - 1) \sum_{v=1}^{\overline{m}} \sum_{i \in S, j \in \overline{S}} x_{ijv}$
(22')

Figure 9 : Hierarchy between constraints (12), (17), (22) and (23).

5. Algorithm

We have developed the following constraint relaxation algorithm for the SDVRP.

PART 1 : INITIALIZATION

Step 1 : Heuristic algorithm. Obtain a first upper bound $\overline{z}$ on the value of the SDVRP solution by applying the heuristic algorithm developed by Dror and Trudeau [7].

Step 2 : Definition of a first subproblem. Define a first subproblem consisting of (1)-(5), (8), the lower and upper bounds on the x_{ijv} variables, as well as a number of constraints of type (18)-(21) and (23).

Step 3 : First subproblem solution. Solve the subproblem using simplex. If the solution is feasible for the SDVRP, the optimum has been reached: stop.

Step 4 : Checking for constraint violations. Check for violations of constraints (13), (22) and (23). If no violated constraint can be identified, insert the subproblem in a stack and proceed to Step 5. Otherwise, introduce a subset of all violated constraints which have been identified, and go to Step 3.

PART 2 : BRANCH AND BOUND

Step 5 : Termination check. If the stack is empty, the optimum has been reached. Print the optimal solution, its value $\overline{z}$, and stop.

Step 6 : Subproblem selection. Select a subproblem from the stack according to a "last in first out" criterion.

Step 7 : Subproblem solution. Solve the subproblem using simplex and let $\underline{z}$ be its solution value. If $\underline{z} \geq \overline{z}$, go to Step 5. Otherwise, check whether the x variables are integer. If so, proceed to Step 9.

Step 8 : Subproblem partitioning. The curent subproblem solution is non-integer. Branch on a fractional x_{ijv} variable, thus creating two new subproblems which are then inserted in the stack. Go to Step 6.

Step 9 : Feasibility check. Check whether the current integer solution contains subtours disconnected from the depot, $k-$split cycles, or vehicle routes whose total demand exceeds the vehicle capacity. Also check for violations of fractional cycle elimination constraints (23). If none can be identified, the solution is then feasible: set $\overline{z} := \underline{z}$, store the solution and proceed to Step 5. Otherwise, introduce a subset of all violated constraints (13), (17), (22) or (23) that have been identified, and to to Step 7. ■

In Steps 2, 4 and 9, several strategies are possible for the generation of violated constraints. (Note that we have chosen to detect violated $k-$split cycle constraints (17) only at an integer solution, in Step 9, in order to simplify the identification process). In Step 2, we experimented with two policies. *Policy 1:* generate no constraint of type (18)-(21) or (23); *Policy 2:* generate all constraints of type (18)-(21) (since their number is only of the order of n) as well as constraints (23) for $v = 1, ..., \underline{m}$ and all pairs (i, j) satisfying $c_{ij} \leq f$, where f is a control parameter. (The value if f is selected so as to avoid generating too many constraints. The value $\underline{m}$ is selected since the number of vehicles in the optimal solution is not yet known, but is certainly at least equal to $\underline{m}$). In Steps 4 and 9, all violated constraints (13) and (17) were generated; in addition, if constraint (23) was violated for some pair (i, j), then it was generated for that pair and for $v = 1, ..., \overline{m}$.

Constraints (13), (17), (22) and (23) are generated as follows. For constraints (13), connected components that include the depot are first identified by means of a labeling procedure. Then S is successively defined as the set of all vertices included in each connected component, excluding the depot, and violations of (13) are then identified for each S in a straightforward manner. For constraint (17), all elementary circuits not including the depot and made up of arcs with positive x_{ijv} values are identified. Then S is successively defined as the vertex set of each such circuits and violations of (17) are identified. For constraints (22), elementary circuits are identified as above and S is successively defined as the vertex set of each circuit and of all unions of circuits with vertices in common. Violations of (22) are then checked for each S. Finally, violated constraints (23) are easily identified by checking in turn all arcs with a positive x_{ijv}.

6. Computational results

In order to gain some insight into the efficiency of the linear relaxation, we carried out a series of computational experiments. We only analyze the behaviour of the algorithm at the root of the search tree (i.e., only Part 1 is executed), as this provides sufficient information on the tightness of the various cuts derived in this paper and we believe the computational burden of developing a full branch and bound scheme (with comparisons of branching criteria, variable fixing procedures, etc.) would add very little to the value of this study. Implementation details of various enumerative algorithm for similar routing problems are provided in the excellent survey of Balas and Toth [3].

All test problems were obtained by using a subset of n demand points of the 75–city problem described by Eilon *et al.* [9]. Note that in this problem, (c_{ij}) is symmetric and satisfies the triangle inequality. (This type of routing problem is usually the hardest). All vehicle capacities were set equal to the same constant $Q = 100$. For each problem, customer demands q_i were generated according to a uniform distribution in $[\alpha Q, \beta Q]$, where α and β are two control parameters. The value of $\underline{m}$ was set equal to $\left\lceil \sum_{i=1}^{n} q_i/Q \right\rceil$, where $\lceil x \rceil$ denotes the smallest integer greater than or equal to x. The value of $\overline{m}$ was taken as n, since star-shaped solutions are always possible, but in practice this bound was never binding. All problems were solved on a SUN 3/50–4 work station.

Five problems were generated for each of the sizes $n = 10$, 15 and 20, with $[\alpha, \beta] = [.1, .5]$. The following values of f were selected after limited experimentation: $f = 25$ for $n = 10$, and 15, $f = 10$ for $n = 20$. In each case, the value $\underline{z}$ of the lower bound provided by LP relaxation was computed after the intial problem solution, for each of the two constraint generation policied, and then, after completion of Part 1. Since the optimal value $z*$ was unknown, we used the ratio $(\overline{z} - \underline{z})/\overline{z}$ to measure the departure of $\underline{z}$ from the optimum. Here, $\overline{z}$ is the initial heuristic solution value. Note that this ration overestimates $(z* - \underline{z})/z*$. The number of pivots and the number of constraints of each type that were generated are reported in Table 1. Additional test were conducted for $n = 10, f = 25$ and various demand distribution parameters $[\alpha, \beta]$. For these problems, we only report the total number of pivots, which is a strong indicator of problem difficulty. The meanings of the various line headings in Table 1 are as follows: GAP(%) : $100(\overline{z} - \underline{z})/\overline{z}$; PIVOTS : cumulative number of

simplex pivots; (13)/(17)/(23): cumulative number of constraints of types (13), (17), and (23) that were generated.

The computational results suggest a number of observations. The various constraints developed for this problem were quite successful in reducing the gap between the lower and upper bounds at the root of the search tree. When constraints (13), (17) and (23) are used in conjunction, the value of the gap is contained between 0% and 9% on all test problems. However, the SDVRP seems considerably harder than the VRP (see for example, Laporte *et al.* [16]) and it appears that branching will almost always be necessary. The low gaps obtained with our algorithm are a strong indication of the quality of the Dror and Trudeau heuristic. Also, our results are consistent with an observation made by these authors [8]: problems with small customer demands tend to require less computational effort (number of pivots) for their resolution. This is shown in Table 2.

7. Conclusion

We have considered in this paper a version of the vehicle routing problem in which split deliveries are allowed. Several families of valid inequalities were developed and a hierarchy between these constraints was established. A constraint relaxation algorithm where branch and bound is used to achieve integrality was then described. It was shown how the various constraints developed help reduce the gap between the lower and upper bounds at the root of the search tree.

Table 1 Computational Results for a Fixed Weight Distribution, and Various Values of n

n	Problem number		After initial LP		At the end of Part 1
			Policy 1	Policy 2	
10	1	GAP (%)	35.66	9.46	5.18
		PIVOTS	249	322	710
		(13/17/23)		(-/-/38)	(5/1/38)
	2	GAP (%)	29.67	2.86	0.87
		PIVOTS	261	349	764
		(13/17/23)		(-/-/38)	(4/2/38)
	3	GAP (%)	37.76	5.75	1.32
		PIVOTS	208	403	1142
		(13/17/23)		(-/-/38)	(5/0/38)
	4	GAP (%)	33.78	8.47	6.01
		PIVOTS	273	313	1242
		(13/17/23)		(-/-/38)	(7/2/38)
	5	GAP (%)	29.15	3.07	0.00
		PIVOTS	216	316	1257
		(13/17/23)		(-/-/38)	(5/1/38)
15	1	GAP (%)	44.98	13.69	2.57
		PIVOTS	863	1360	9959
		(13/17/23)		(-/-/86)	(22/3/86)
	2	GAP (%)	43.33	17.21	8.38
		PIVOTS	840	1312	8642
		(13/17/23)		(-/-/86)	(25/2/86)
	3	GAP (%)	35.00	10.89	4.61
		PIVOTS	736	1516	6565
		(13/17/23)		(-/-/86)	(19/1/86)
	4	GAP (%)	33.75	10.42	7.15
		PIVOTS	813	1098	3687
		(13/17/23)		(-/-/86)	(10/2/86)
	5	GAP (%)	47.53	15.31	1.47
		PIVOTS	801	1586	18357
		(13/17/23)		(-/-/86)	(42/5/86)

Table 1 (Continued) Computational Results for a Fixed Weight Distribution, and Various Values of n

n	Problem number		After initial LP		At the end of Part 1
			Policy 1	Policy 2	
20	1	GAP (%)	44.46	25.46	8.74
		PIVOTS	1387	2145	29112
		(13/17/23)		(-/-/16)	(39/9/176)
	2	GAP (%)	41.06	20.89	5.25
		PIVOTS	1575	1768	36817
		(13/17/23)		(-/-/16)	(42/8/116)
	3	GAP (%)	43.98	24.81	4.64
		PIVOTS	1468	1500	35454
		(13/17/23)		(-/-/16)	(47/10/176)
	4	GAP (%)	47.09	25.27	3.85
		PIVOTS	1612	2421	28709
		(13/17/23)		(-/-/16)	(43/11/196)
	5	GAP (%)	50.17	19.63	0.74
		PIVOTS	1602	1893	23727
		(13/17/23)		(-/-/16)	(37/6/176)

Table 2 Total Pivot Count for 5 Problems, $n = 10$, and Six Weight Distributions

	Weight distribution					
Problem number	[.01, .1]	[.1, .3]	[.1, .5]	[.1, .9]	[.3, .7]	[.7, .9]
1	804	855	710	3138	3190	8570
2	1236	1384	764	3961	3604	9488
3	1145	953	1142	5251	6816	2200
4	812	620	1242	4819	4331	2193
5	1200	561	1257	3496	4701	3118

References

[1] K. Altinkemer and B. Gavish, Heuristics for delivery problems with constant error guarantees, Transportation Science 24 (1990) 294–297.

[2] K. Altinkemer and B. Gavish, Heuristics for unequal weight delivery problems with a fixed error guarantee, Operations Research Letters 6 (1987) 149–158.

[3] E. Balas and P. Toth, Branch and bound methods, in The Traveling Salesman Problem. A guided Tour of Combinatorial Optimization, E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan and D.B. Shmoys (eds.), Wiley, Chichester (1985) 361–401.

[4] N. Christofides, Vehicle routing, in the Traveling Salesman Problem : A guided Tour of Combinatorial Optimization, E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan and D.B. Shmoys (eds.), Wiley, Chichester (1985) 431–438.

[5] G.B. Dantzig, D.R. Fulkerson and S.M. Johnson, Solution of a large scale traveling-salesman problem, Operations Research 2 (1954) 393–410.

[6] M. Desrochers, Private communication (1989).

[7] M. Dror and P. Trudeau, Savings by split delivery routing, Transportation Science 23 (1989) 141–145.

[8] M. Dror and P. Trudeau, Split delivery routing, Naval Research Logistics 37 (1990) 383–402.

[9] S. Eilon, C.D.T. Watson-Gandy, N. Christofides, Distribution Management : Mathematical Modelling and Practical Analysis, Griffin, London (1971).

[10] B. Gavish, The delivery problem: New cutting planes procedures, Presented at the TIMS XXVI Conference, Copenhagen (1984).

[11] B. Gavish, Augmented Lagrangean based algorithms for centralized network design, IEEE Transactions on Communications 33 (1985) 1247–1257.

[12] G. Laporte, Generalized subtour elimination constraints and connectivity constraints, Journal of the Operational Research Society 37 (1986) 509–514.

[13] G. Laporte, The vehicle routing problem: an overview of exact and approximate algorithms, European Journal of Operational Research (1992), forthcoming.

[14] G. Laporte and Y. Nobert, A branch and bound algorithm for the capacitated vehicle routing problem, OR Spektrum 5(1983) 77–85.

[15] G. Laporte and Y. Nobert, Exact algorithms for the vehicle routing problem, in Surveys in Combinatorial Optimization, S. Martello, G. Laporte, M. Minoux and C. Ribeiro (eds.), Annals of Discrete Mathematics 31, North-Holland Publishing, Amsterdam (1987) 147–184.

[16] G. Laporte, Y. Nobert and M. Desrochers, Optimal routing under capacity and distance restrictions, Operations Research 33 (1985) 1050–1073.